THE

JACG NEWSLETTER

JACG

THE JERSEY ATARI COMPUTER GROUP

($2.00)

VOLUME 7 NUMBER 9 BBS : 201-298-0161 NOVEMBER 1987

E D I T O R I A L

Please note the "new" front, back, and inside back pages which have been supplied by Linda Peckham via desktop publishing. Also please note the pagination supplied by Steve Godun. I will be experimenting with various formats, et al., in the next issues, comments, suggestions and submittals will all gladly be received.

I had to go to press earlier than usual this month (although "earlier" was still later than the "official" cut-off date of the 20th of the month) as I will be on a business trip during the weeks prior to, and following the meeting.

To set the record straight...it was mentioned by a member at the last meeting, that GEMINI of Cedar Knolls was no longer going to support the ATARI 8-bit machines. Bob, at GEMINI, states that this simply is not the case; and that perhaps his policy in doing business with public schools was mis-interpreted as a policy regarding the ATARI 8-bit line. A visit to GEMINI will assuage your fears... almost 1/2 the software there is 8-bit ATARI...I mean, after all, as long as there is an active clientelle...c'mon folks.

FLASH

A seperate mailing has, or will be made of a form that MUST be filled-out by ALL AT&T employee/members. It is essential that this form be returned as soon as possible. Why? In order for us to use the meeting facilities at Bell Labs in Murray Hill, we (and it makes sense) must continue to have a portion of our membership employees of AT&T.

I hope that I am current with your Newsletter submissions. I think that I am. If I am not, please get hold of me and let me know of my lapse.

I hope everyone has a happy and bountiful "turkey-day", may your ATARI cornucopia overflow...'til next month...

IN THIS ISSUE

Let the JACG - G. Gorski..... 3

Flea Market Rules - JACG..... 3

Prsident's Corner - T. Pazel..... 3

From Jay's Perch - J. Abrams..... 4

Word Magic - P. Greenhut..... 5

More Power 2U - M. Kiss..... 6

Warp Power Two - M. Kiss..... 7

Limericks for Engineers - D. Forbes.. 9

From the Desktop - L. Peckham..... 10

PDG - D. Van Hook..... 12

Where is Our Support? - S. Godun..... 13

Longer Ribbon Life - N. Van Dost, Jr. 14

Noise from Noyes - D. Noyes..... 14

Atarifest, D.C. - L. Peckham..... 15

My First Robot - D. Forbes..... 16

MARK YOUR CALENDAR !!
JACG
MEETING SCHEDULE
DECEMBER 12, 1987

EFFECTIVE 11/1/87

ATARI® ST Computer System

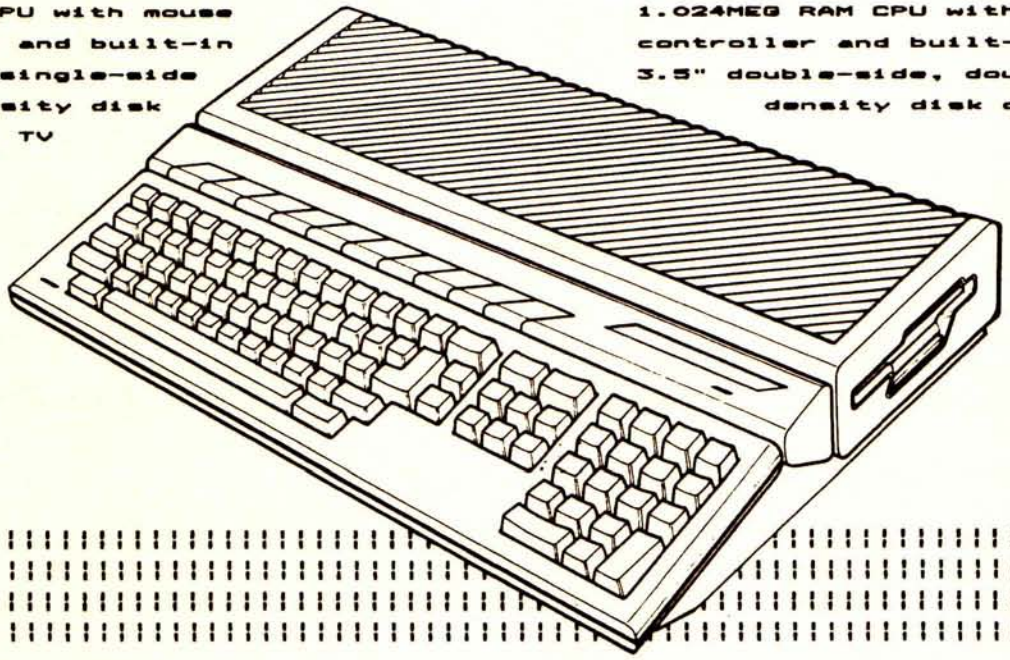
68000 MICROPROCESSOR - 8 MHZ CLOCK - RS232 SERIAL PORT - PARALLEL PORT
HARD DISK DRIVE PORT - CARTRIDGE PORT - SECOND FLOPPY DRIVE PORT - 192K ROM
MIDI INTERFACE - TOS OPERATING SYSTEM - GEM DESKTOP - 512 COLORS
MONOCHROME 640 X 400 RESOLUTION - COLOR 640 X 200 RESOLUTION

90-day Warranty - Over-the-Counter Exchange on DOA Defects

FREE Word Processor with system purchase (limited quantity)

520STFM.....$ 599.95
512K RAM CPU with mouse controller and built-in 360K-3.5" single-side double-density disk drive plus TV output

1040STF.....$ 799.95
1.024MEG RAM CPU with mouse controller and built-in 720K 3.5" double-side, double-density disk drive



MONO SYSTEMS

520STFM CPU with .....$ 699.95
ATARI SM124 Hi-Res B&W Monitor

1040STF CPU with.....$ 899.95
ATARI SM124 Hi-Res B&W Monitor

COLOR SYSTEMS

520STFM CPU with .....$ 799.95
MAGNAVOX CM8505 Lo-Res RGB Color Monitor

1040STF CPU with.....$ 999.95
MAGNAVOX CM8505 Lo-Res RGB Color Monitor

520STFM CPU with .....$ 899.95
ATARI SC1224 Med-Res RGB Color Monitor

1040STF CPU with.....$1099.95
ATARI SC1224 Med-Res RGB Color Monitor

GEMINI ENTERPRISES
86 Ridgedale Avenue
Cedar Knolls, NJ 07927

(201) 267-0988

LET THE JACG
(Do The Work For You)

G. Gorski - JACG

Sit back, relax, and earn extra money, by letting the JACG sell your ORIGINAL software. For one reason or another, we all buy software that just isn't right, has been outgrown, or is no longer needed.

The JACG will sell that software (maximum of 3 per month) at its' monthly meetings. All you have to do is to bring in the software and ORIGINAL documentation, clearly marked with: your name, member number, and the price you would like to sell it for!! For 20% of the sale price, the JACG will handle the sale for you, with no hassles. Please be sure to bring your software to the meeting early, so that it can be displayed at the flea market. This can be a wonderful way to get that much needed extra ca$h. If you need additional information please call, or drop me a line:

Gary Gorski
313 Sheridan Ave.
Roselle,NJ 07203
201-241-4554

****
FLEA MARKET RULES
****

In order to clarify the intention of the Executive Committee in sanctioning the use of the BTL lobby before and after monthly meetings for use as a member flea market we publish the following rules:

1. All flea market sellers must be current JACG members.

2. Space is provided on a first-come, first-served basis.

3. Only ORIGINAL programs <u>with</u> ORIGINAL documentation may be sold in the area of software.

4. Hardware of any type may be sold normally without constraint. The Executive Committee reserves the right, however, to limit the physical size and space consumed by such hardware.

5. Flea market business will be conducted only in the lobby and ONLY when the meeting is not in session in the auditorium.

6. The Executive Committee reserves the right to deny or suspend the privilege of flea market usage to any person, member or not, for infraction of these operating rules.

PRESIDENT'S CORNER

Tom Pazel - JACG

Well, I finally got my first "in person" look at the new Mega STs the other day. First impression? Pretty nice looking machine. Obviously very powerful CPU-wise and graphically. I hope they sell well. They should. With the hundreds of programs currently available for them, as well as the ability to "run" MS/PC-DOS software (via "PC-Ditto"), I can see the Mega ST quickly becoming a strong runner in the small business/home power user market. I think one of the keys to their success lies in the marketing strategy taken by Atari with these machines. I don't know how they plan to handle this, as I haven't seen any ST commercials yet. I am looking forward to seeing what happens in this situation.

I wish I had a lot more to say this month, but I'm afraid I don't. I've been rather distant from the Atari world lately, not by my own choosing. This being my final column as president, I must say that I've enjoyed my reign over the JACG. I think all of us have accomplished quite a bit in the past year. I believe there is more to be done and that all our goals as a group can and will be met.

Again, I thank you all for having me. Let me leave you with this thought: You are a member of this club. Let your voice be heard. If there is something you don't like (or something you do), TELL SOMEONE in the Executive committee. If <u>you</u> don't say it, nobody might. This organization (as with most) won't get any better without ideas and communication.

I wish you all the best!

FROM JAY'S PERCH

Jay Abrams - JACG

In one of the boldest moves in its history, Atari will introduce a new 32 bit computer at this coming Comdex. Contrary to expectations, this machine, often dubbed the "TT", will NOT be based on the Motorola 68020 but will use the Inmos T-800. The T-800 is extremely new and incredibly powerful: it is TWELVE TIMES faster than the combined Intel 80386 and 80287 chips! But the T-800 is much more: this chip (also known as the Transputer) can be used in groups to create multiprocessor or parallel processing computers.

Several features make the T-800 (also known as the Transputer) special. This microprocessor was one of the first RISC chips, RISC standing for reduced instruction set an approach to chip design that allows simplification and reduced size and consequently much, much greater speed. For even more speed, the T-800 variant of the Transputer has a floating point processor built right onto the chip. This processor works concurrently with the main processor. Crunching numbers fast isn't enough, you've got to have your instructions and data handy, consequently the T-800 has 4K of super fast static RAM which is five times faster than normal system RAM. And more! Ultra fast serial communications lines permit the Transputers to form arrays of processors. And these computer arrays work: Meiko a British computer company has built a 3000 MIPS (Million Instructions Per Second) using 360 Earlier model Transputers.

From a commercial standpoint, the Transputer alows a company to build a whole family of computers from the micro to the super scale. Atari will be the FIRST mass production computer company to utilize this chip.

Despite the great power of the T-800, Atari is also adding ASICs (Application Specific Integrate Circuits) of its own design to the new machine. These machines will have the thousand by thousand resolution monitor Atari promised last year.

To develop the operating system for this computer, Atari teamed with a British company Perihelion Software. The resulting operating system is called Helios. Like UNIX it will probably be multiuser and multitasking, however UNIX was not intended for massively parallel computers as the Transputer machines most certainly will be. Atari also retains its traditional interest in graphics which was not one of the strong points of UNIX.

If British journalists are correct, Perihelion wants Helios to be another MS-DOS and Atari will not have exclusive use of the system despite the fact that it is a part owner of the company. Another interesting fact: Helios is being developed under the supervision of Tim King who developed AmigaDos!

In many ways, the "TT" is as risky a project as the ST. The ST required custom chips and those chips pretty much had to work the first time around or Atari was out of business. The "TT" is using a chip that had been commercially available for only a few weeks at the time this column was written! And Atari wants to be in production six months from now!

Atari's preoccupation with the "TT" explains its relatively conservative actions of the last twelve months. If you think about it, most of the product that appeared was repackaged or enhanced versions of earlier work. In the case of the Mega ST, the enhancements were significant, and the product is quite important to Atari's future. More on the Mega in my next column.

Material for this column came from ELECTRONICS, PERSONAL COMPUTER WORLD, BYTE, INFOWORLD, and THE NEW YORK TIMES.

◆

WORD MAGIC

Philip Greenhut - JACG

THE SUPERIOR WORD PROCESSOR AND MUCH MORE

Word Magic by Jim Thompson

Blue Collar Software has a motto that "IT WORKS." I am here to tell you that this little package (that can be purchased from Gemini or Antic) gives you Word Magic, Spell Magic and Graphic Magic and really does work! It does it all and more and all for under $25. It is definitely the best Atari word processor on the market. Try it and you will be convinced.

In fact it just received an excellent review in Analog.

It has a disk side of instructions which can be printed out for reference. It is fully designed to take advantaged of the RAM Disk.

Version 5 for the 130XE is different from earlier versions in being able to use Data Disks without any program files being put on them. All you need is a blank unformatted disk. Word Magic will format it and you can cram it with only data files.

When we load our program disk we get three choices which we can select:
Word Magic, Graphic Magic and a super Spell Magic with an unlimited customised personal dictionary which can take over a million words of your choice. All three programs are intergrated with one another. We pick Word Magic from the menu. It comes up and we have a simple menu with help screens. You won't need the help screens unless you are a novice.

Word Magic allows you to do several things from its master menu. One keystroke-usually the first letter of each function-will access most of the following functions:

Edit File, Load File, Save File, Print File, View File, Directory List, Delete File, Format Disk, Unlock File, Magic Program Menu.

One of its unique features is two POP UP SCREENS for text editing.

First hit SELECT to get the first POP UP SCREEN. Then Hit B to go to the bottom of your file; T for the top; L for left of line; R for right of line. U for Up Scroll and D for down scroll. By the way, you can set up three different scroll speeds if you like. Those of you who love speed will love this little benefit.

The second POP UP SCREEN really separates WORD MAGIC from other word processors. Hit Option and you get Search, Replace, Global Search & Replace, Next Search Occurance, Block Function and Include File. Hit the first letter of any of these and you are in the function. If you select Block Function you will then be able to highlight the block of text that you want to use. After you do this using the cursors you then pick one of the following:

A. Delete text selected.
B. Move text selected
C. Copy text selected
D. Write text selected

Some of the above are practically instantaneous like Delete. Some like Move may take a little longer if you wish to Move to the middle of a long file.

Another nice feature is by using control codes you can Right Justify, Left Justify, Justify Both Sides and finally Center.

I always like to save something major to disk before I do anything else. Now just by hitting S again we can Save this in Ram disk. Instead of trying to put in an exact file name I put in one letter like A for the file name in Ram Disk.

Now I can go in and out of RAM Disk at will. If I want to print I hit P and you will see how quick the file loads into Print mode as opposed to using D:1. Now if you like you can change any of your prearranged print parameters or just start printing. If you have long files and a high capacity print buffer this should load in a split second and enable you to do other jobs while printing.

One of the nice things is that by using Control D after most of the regular print format codes you will get NLQ(NEAR LETTER QUALITY). Try it you will find it really works.

You can use Epson Compatible and Gemini Printers from preloaded print programs. Another program allows you to set to most other printers. Just follow the default menu

the first time you use the program. After that you can adjust just before printing if you dont like your default. In fact this default menu controls alot of functions all of which can be changed later and very easily.

In the above paragraph we have tried to show some of the print attributes of this program.

Word Magic works without a lot of disk switching on one drive. Two drives improves the operation particularily with Spell Magic.

So, we have an easy menu driven word processor and I think those of you that use Word Magic will agree that it is easy to use and advanced enough for just about any user.

In addition we have Graphic Magic to integrate pictures or graphics with our word processor and also super Spell Magic.

If you have any questions regarding this program please feel free to call me at 201-925-8160 days or 201-992-4928 nights till 10.

IT WORKS try it you will love it.

Jim Thompson is now writing for the ST and is looking for someone to take over where he has left off with Word Magic. The Source Code is in ATARI MACRO ASSEMBLER. If you are interested please contact the above.

◆

MORE POWER 2 U

Marvin Kiss --JACG

The article in the July, 87 issue of the JACG newsletter by Kenneth J. Pietrucha was of interest to me because I, too, have been bitten by the multidigit bug. Looking over his program to calculate all the digits of large powers of 2 I noticed that it did not do justice to the power of Atari Basic.

Each element of array A is capable of holding integers up to 9999999999 with no loss in precision. Therefore each can hold ten digits of the sum yielding a speedup of about ten to one. OF course, the test for a carry to the next element must be performed before the addition to prevent exceeding ten digits. Packing more digits also allows us to work with up to 40,000 digits in an array dimensioned only 4000.

This change is not achieved without increased complexity in the printout since we must assure that each element (except the first) produces ten printed digits even when the most significant are zero. One method of implementing this is shown in the following listing on line 60. The array element is first converted to a string in D$, the length is found, and then the string is used to replace the last digits of a string of ten zeros.

Many other time saving tricks were used to speed execution. Most take advantage of known idiosyncrasies of Atari Basic:

1. Define variables for constants used repeatedly. This allows the constant to be interpreted once instead of each time it is encountered.

2. Don't use GOTO's in loops if they can be avoided. If a GOTO is encountered, Basic searches from the beginning of the program for the destination line number. (A similar search occurs when a NEXT causes a branch to the corresponding FOR.)

3. Take advantage of multiple NEXT statements. Look at lines 20 and 30. They both contain NEXT I and NEXT M but only one NEXT will be executed for any particular iteration. If this logic were implemented in the normal way, the IF on line 30 would end in a GOTO to branch around line 40 to a single NEXT I: NEXT M, wasting valuable time.

4. Take advantage of known conditions. Again in line 30 if the I loop terminates we know that a carry was generated so we can immediately extend the array and put in a leading 1.

5. Don't throw away useful results. Lines 120 and 130 make use of the previous answer if the desired power of 2 is equal to or greater than the previous one.

6. Put multiple statements on one line. Basic does extra work whenever a line number is encountered.

7. Searches for GOTO and NEXT targets are also improved if there are fewer lines and the destinations are near the beginning of the program. Put the most used statements nearest the beginning of the program and the least used last.

8. Shutting down the display while doing lengthy computations makes sense, so this feature has been retained. Added is a sound to alert a resting operator that the answer has been found.

9. Eliminate unnecessary work. There is no need to initialize the answer array since each added element is set to ! as needed. The answer is now printed only when the screen is on, saving the time to

convert answers to display digits except when necessary.

What is the result of all this foolery? The time to compute 2 to the 1000 has been reduced from the 2 and 1/2 hours reported by Ken to about 3 and 1/2 minutes. Since the running time is proportional to N squared the time to compute 2 to the 132049 would take about 5 years with the original program or about 43 days with this version.

Even this running time is much too long to be practical, so next time I will present a program that runs at WARP speed.

```
10 T=10:DIM D$(T),P$(T),Z$(T),A(4000):GOTO 200:REM "MOREP2
  .BAS"
20 FOR M=S TO N:FOR I=Z TO D:B=A(I):IF B<MAX THEN A(I)=B+B
  +C:C=Z:NEXT I:NEXT M:GOTO 50
30 B=B-MAX:A(I)=B+B+C:C=WON:NEXT I:D=D+WON:A(D)=WON:C=Z:NE
  XT M
50 POKE 559,34:PRINT A(D);
60 IF D THEN FOR I=D-WON TO Z STEP MWON:D$=STR$(A(I)):P$=Z
  $:P$(EL-LEN(D$))=D$:PRINT P$;:NEXT I
70 L=LEN(STR$(A(D)))+T*D:PRINT :PRINT "NUMBER OF DIGITS =
  ";L
80 SOUND 0,100,10,15:FOR I=0 TO 50:NEXT I:SOUND 0,0,0,0:S=
  N
90 PRINT " 2 TO THE POWER OF N":PRINT " ENTER VALUE OF N "
  ;:INPUT N
100 IF N<Z THEN PRINT "POSITIVE N ONLY":GOTO 90
110 IF N<S THEN A(Z)=WON:D=Z:S=Z
120 IF N=S THEN 50
130 POKE 559,Z:S=S+WON:GOTO 20
200 Z$="0000000000":MAX=5000000000:Z=0:WON=1:MWON=-WON:EL=
  T+WON:S=Z:D=Z:A(Z)=WON:GOTO 90
```

◆

WARP POWER TWO

Marvin Kiss --JACG

In my previous article I attempted to produce all the decimal digits of high powers of 2 as quickly as possible in pure Atari Basic. But Atari Basic must interpret each instruction every time it is executed, making it incredibly slow compared to a compiled language, such as Fortran. In this article, I will try to unleash the full power of the 6502 microprocessor on the same problem.

Unfortunately, I know of only one way to do number crunching at maximum speed. It involves Assembly Language programming. Fortunately, a very small part of our Basic program does almost all the work. A fairly short Assembly Language subroutine which adds a string of digits to itself, callable from Basic, can be written. I will first discuss the Assembly routine, and then the Basic program.

The 6502 has a decimal mode which supports adding bytes containing two decimal digits each. This is a natural for our task. To make the answer accessible to Basic, we plan to dimension a string variable in Basic to be used as the byte array by our subroutine.

The Assembly Language subroutine, WARP2.ASM, has two entry points:

1. INIT at address 1536 (HEX 600) has one argument which supplies the string address.

2. POW at address 1566 (HEX 61E) also has one argument of minus the iteration count. The use of a negative count makes the subroutine simpler. The Basic USR call is limited to 16 bit arguments, so the maximum number that can be used is 65535 which is equivalent to minus one. If the argument is zero, the number of iterations will be 65536.

I will give the execution counts for various sections of the program based on computing 2^132049 so you can appreciate the importance of careful coding of the most used parts.

When trying to achieve the highest speed, I usually start by writing the innermost loop, lines 440 - 480. These five instructions are executed 1.3 billion times, so any time saved here will really pay off. The next part, 500 - 550, is needed because the index registers are only one byte long. Every 256 bytes, or 512 digits, we must adjust the high order part of the string address (5 million times).

When the string has been doubled, the instruction at 570 checks for a carry. If true, instructions 590 - 780 lengthen the string by one byte, set the new byte to 1, and make adjustments to the starting index values and addresses for the next iteration of the main loop (20 thousand times). The main loop iteration counter is incremented at lines 790 - 820 and tested for zero. If not zero, the instructions in lines 260 - 390 set up the index registers and addresses used in the innermost loop (132 thousand times), and away-we-go!

When the main loop iteration count is exhausted, lines 840 - 920 return the (string length -1) to Basic as the USR function value.

Continuing in my backwards explanation, lines 200 - 240 get the power iteration count from Basic and store the count in low memory where it takes less time to access. Now, to end at the beginning. Lines 60 - 180 provide the initialization by getting the string address from Basic, setting the string length to 1, and then jumping to line 600 to make use of the same calculations that are used when the string is lengthened. (I may be lazy but this saves a lot of duplication.)

Its easy to see from the iteration counts that over 90% of the execution time will be in the innermost loop, lines 440 - 480. I have attempted to do this as efficiently as possible. The dummy $1234 address in the first three instructions is replaced during execution with an actual address calculated each time the string is

lengthened and modified (high order part) every 256 iterations. Programming purists will frown on the practice, but we pragmatists use whatever works! It would have been easier to use a two byte pointer on page zero, but it would have taken more execution time.

When I wrote the first version of WARP.ASM, I did not include the main loop. In testing, I found that it took 6 seconds to do 2^1000 with the main loop in Basic. Then I tried executing a USR function that did nothing but return, and it took 5 seconds for 1000 iterations! That's why I put the main loop into the Assembly routine. This choice makes the running time for 1000 iterations look good, but it shaves only 11 minutes from the time for our example because this loop is used only N times, not N^2.

For those few of you who are not familiar with Assembly Language, let me explain that the source code presented here is next processed by an Assembler which converts it to a Machine Language object file in a form recognizable to a Binary Loader program. The file contains the load address, the byte count, and the bytes to be loaded into memory. By convention, if the source file name is WARP2.ASM, we name the binary file WARP2.OBJ if it is a subroutine, or WARP2.EXE if it is a stand-alone program.

The Basic program is much like the version I presented in the previous article. The problem with Assembly subroutines is that there is no instruction in the original Atari Basic to automatically load an object file from disk. (The XL version now has an XIO function to perform this task.) A trick I use is to use XIO 13 (status check) to put the file name into the IO system and then jump to the binary file loader with a four byte USR subroutine (see lines 210-240). This procedure works with Atari DOS 2.0S. To prevent unnecessary disk operations, I check in line 200 for the presence of an instruction at the POW entry address to determine if the subroutine is already loaded.

There are two other easy ways to load a binary file: the DOS MENU load binary file function, L, or by naming the file AUTORUN.SYS which causes it to be loaded when the disk is booted.

Since each byte of string P$ can hold 2 digits, the 20000 dimension provides for up to 40000 digits. It is very important that all strings and arrays be dimensioned before the address of the string P$ is found. This is required because Basic sometimes moves strings around in memory, but we need the address to remain fixed. The assignment of a character to P$(20000) in line 10 is needed to establish the string length so Basic will not detect an error when we use the ASC function in lines 50 and 60.

In converting the string data to decimal notation for printing, we cannot simply print the numeric value of the ASCII code since each digit occupies its own four bits. They must be separated by dividing by 16 instead of the usual 10.

Now for the good news. The time to do 2^1000 has been cut to 1 second. This means that our goal of 2^132049 is doable in about 5 hours!!!

One important problem with assembly language merits mention here. That is the question of correctness. How do we know that the answer is right? No matter how fast we can calculate, what is it worth if we get the wrong answer? With the original Basic program we are relatively sure the answer is right; but, the more tricks we apply and the more obscure the code (Assembly language), how can we be sure? The only suggestion I have is to use two independent programs, preferably on different machines, and in different languages, and if the answers agree, so be it.

One more word of caution. In checking results, you must take into account the special cases. The WARP.ASM program executes different code when the string length is above 256 bytes, so answers above and below 512 digits must be checked for consistency. It also can do only 65536 iterations with one call. So, if we want to try to do more than that in one fell swoop, we should check that the logic in line 20 of the Basic part is correct.

To save you countless hours of typing, I will give a copy of these programs to the JACG library. Good luck.

```
10 ;WARP2.ASM BY MARVIN KISS--JACG
20   *=$600     SET ORIGIN AT PAGE 6
30 COUNTL=$CB    DEFINE ADDRESS FOR
40 COUNTH=$CC     MAIN LOOP INTERATION
50 ;              COUNTER
60 INIT PLA     INITIALIZATION ENTRY
70  PLA         GET ADDRESS OF STRING
80  STA STADH      HIGH BYTE
90  STA LAST+2
0100  PLA
0110  STA STADL      LOW  BYTE
0120  STA LAST+1
0130  LDA #$FF     INITIALIZE FOR
0140  STA LX+1       LENGTH OF ONE
0150  STA LY+1       BYTE
0160  STA COUNTH  AND TO STOP MAIN
0170  STA COUNTL         LOOP
0180  JMP PUT1     JUMP TO PUT1
0190 ;-----------------------------
0200 POW PLA    RUN ENTRY POINT
0210  PLA           GET ITERATION
0220  STA COUNTH      COUNT FROM
0230  PLA             BASIC
0240  STA COUNTL
0250 ;----------BEGIN MAIN LOOP------
0260 AGAIN LDA #0   GET LOW BYTE OF
0270 ADL=*-1          ADDRESS
0280  STA L1+1          AND STORE
0290  STA A1+1          WHERE NEEDED
0300  STA S1+1
0310  LDA #0        GET HIGH BYTE OF
0320 ADH=*-1           ADDRESS
0330  STA L1+2          AND
0340  STA A1+2          STORE
0350  STA S1+2
```

```
0360 LX LDX #0  LOAD INDEX REGISTERS
0370 LY LDY #0  WITH MINUS STR LENGTH
0380  SED        SET DECIMAL MODE!!
0390  CLC        CLEAR CARRY
0400 ;-------------------------------
0410 ; HERE IS WHERE WE ACTUALLY ADD
0420 ; !!! BEGIN MOST USED LOOP !!!
0430 ;-------------------------------
0440 L1 LDA $1234,X GET BYTE,2 DIGITS
0450 A1 ADC $1234,X   ADD TO ITSELF
0460 S1 STA $1234,X   PUT BACK RESULT
0470  INX             INCREMENT X
0480  BNE L1   IF X NOT ZERO GOTO L1
0490 ;----------END OF MOST USED LOOP
0500  INY            INCREMENT Y
0510  BEQ IDN      IF Y ZERO GOTO IDN
0520  INC L1+2       INCREMENT HIGH
0530  INC A1+2         ADDRESS
0540  INC S1+2         BYTES
0550  JMP L1      AND CONTINUE TO ADD
0560 ;----------END OF STRING ADD
0570 IDN BCC FIN    IF CARY NOT SET
0580 ;                JUMP TO FIN
0590 ;-----ADD ONE MORE BYTE TO STRING
0600  LDX LX+1   ADJUST LOW BYTE OF
0610  BNE ID2        STRING LENGTH
0620  DEC LY+1     AND H IF NECESSARY
0630 ID2 DEC LX+1

0640  INC LAST+1   ADJUST ADDRESS
0650  BNE PUT1       OF END OF STRING
0660  INC LAST+2
0670 PUT1 LDA #1     PUT A 1 IN THE
0680 LAST STA $1234    NEW DIGIT
0690  CLD             SET BINARY MODE
0700  SEC             SET CARRY
0710  LDA #0         GET STRING START
0720 STADL=*-1        ADDRESS
0730  SBC LX+1        SUBTRACT INDEX
0740  STA ADL         VALUES TO FORM
0750  LDA #0          BASE ADRESS
0760 STADH=*-1        AND SAVE FOR
0770  SBC #0            NEXT
0780  STA ADH           ITERATION
0790 FIN INC COUNTL  INCREMENT MAIN
0800  BNE AGAIN       LOOP COUNTER
0810  INC COUNTH        AND LOOP BACK
0820  BNE AGAIN         IF NOT ZERO
0830 ;-------------- DONE ---------
0840  CLD             SET BINARY MODE
0850  SEC             SET CARRY
0860  LDA LAST+1        GET (LENGTH
0870  SBC STADL         OF STRING)-1
0880  STA $D4           AND RETURN
0890  LDA LAST+2        VALUE TO
0900  SBC STADH         BASIC
0910  STA $D5
0920  RTS             RETURN TO BASIC
0930  .END
```

```
10 DIM P$(20000):P$(20000)="":Z=0:WON=1:ST=16:M=65536:NP=
  1566:INIT=1536:SCR=559:GOTO 200:REM "WARP2.BAS"
20 IF N-S>M THEN V=USR(NP,Z):S=S+M:GOTO 20
30 V=USR(NP,M-N+S):S=N
40 POKE SCR,SCON
50 L=WON:Y=ASC(P$(V+1)):IF Y>9 THEN L=TWO:Q=INT(Y/ST):PRIN
  T Q;:Y=Y-Q*ST
60 PRINT Y;:IF V THEN FOR J=V TO WON STEP MWON:Y=ASC(P$(J)
  ):Q=INT(Y/ST):R=Y-Q*ST:PRINT Q;R;:NEXT J
70 PRINT :PRINT "NUMBER OF DIGITS = ";L+TWO*V
80 SOUND 0,100,10,15:FOR I=0 TO 50:NEXT I:SOUND 0,0,0,0
90 PRINT " 2 TO THE POWER OF N":PRINT " ENTER VALUE OF N "
  ;:INPUT N:POKE SCR,Z
100 IF N<Z THEN PRINT "POSITIVE N ONLY":POKE SCR,SCON:GOTO
   90
110 IF N<S THEN V=USR(INIT,PA):S=Z
120 IF N=S THEN 40
130 GOTO 20
200 TWO=2:SCON=34:MWON=-1:IF PEEK(NP)=104 THEN 240
210 DIM L$(4):FOR I=1 TO 4:READ X:L$(I)=CHR$(X):NEXT I
220 DATA 104,76,164,21
230 XIO 13,#1,4,0,"D:WARP2.OBJ":B=USR(ADR(L$))
240 S=Z:PA=ADR(P$):V=USR(INIT,PA):GOTO 90
```

LIMERICKS FOR ENGINEERS

Donald Forbes - JACG

There was a mathematical dame,
Who sought a prime number for fame.
  She bought a computer,
  Then served as its tutor,
And the number she wanted soon came.

If you are seeking a course that is fun,
To the registrar walk do not run.
  For logic and sets
  And the odds on your bets,
The right choice is Math one-oh-one.

A mathematician named Hugh
Programmed his cat to say "mu."
  All the cats in a "rho"
  Were there "delta" blow
For this cat "nu" more than they "nu."

A simulationist was he,
Never dealing in reality.
  For all occasions,
  He had equations,
To describe how the world should be.

The derivative of cosine
Is the same as minus the sine.
  Thus, its value at pi
  Plus one half six i
Is a square root of negative nine.

(Thanks to T. Ansler, G. Markus, D. Moore, C. Newberry, D. Teets and SIAM News.)

FROM THE DESKTOP

BY Linda Peckham
917A Preakness Avenue
Wayne, NJ 07470
201-790-3061

Greetings, and welcome to another column! This month's article contains a number of items, including ...

MORE FONTS!

SCHOOLBOOK-- abcdefghijklmnopqrstuvwxyzABDCEFGHIJKLMNOPQRSTUVWXYZ0123456789!!@#$%^&*()[]{};:'",./<>?

DINGBATS-- ✽✽✽●○■□□□□▲▼◆◆◗||✍ ✽✽✽●○■□□□□▲▼◆◆◗||✍ ✽✽✽✽✽●✐∞➡✓✓✗✗✗✚✂✂✉ ✂✂✉✆✍✈✉✳✳✆✆✚✚✆✂✌✎✏ ✥✝✝

SYMBOLS -- αβχδεφγηιφκλμ-νοπθρστυϖωξψζΑΒΔΧΕΦΓΗΙϑΚΛΜΝΟΠΘΡΣΤΥςΩΞΨΖ0123456789!!≅#∃%⊥&*()[]{};:∍∀,./<>?

COMPUTER -- abcdefghijklmnopqrstuvwxyzABDCEFGHIJKLMNOPQRSTUVWXYZ0123456789!!@#$%^&*()[]{};:'",./<>?

HUDSON -- abcdefghijklmnopqrstuvwxyzABDCEFGHIJKLMNOPQRSTUVWXYZ0123456789!!@#$%^&*()'"'";:'",./,.?

SATURN -- abcdefghijklmnopqrstuvwxyzABDCEFGHIJKLMNOPQRSTUVWXYZ0123456789!!@#$%^&*()[]{};:'",./<>?

FIGURE 1. The screen image of the upside-down and slanted "JACG." Compare the slant, to the slant on the front cover.

The first four fonts are on disks 4 and 5 from Font Factory. The last two are Public Domain, from Soft Logik. They, along with a new version of the Postscript driver, will be in the library.

DROPPING A CAP, PART II

When experimenting, it helps to re-read the documentation -- and any how-to articles. There may be an easier way to do a certain function. Case in point? How to drop a capital. The easiest way to do that is not with either method in the last column, but as follows:

The large capital will be a separate text object. Go to the text mode, and select the lines that will be indented by the cap. Then, hit Alt-M. This brings up the Margins dialog box. Adjust the left margin, click "OK", and voila! Of course, you'll still need to adjust the capital to the correct position.

This method, along with other hints, is published in a new ST magazine from *Canada*, called *RESET*. It will be published quarterly, and the emphasis will be on how-to articles, and reviews. If you are not into the programming and technical orientations of magazines like *START* or *ST APPLICATIONS*, then this may be a magazine to look into.

BUGS BUGS BUGS

Has anyone else noticed this bug? While working on a PP version of the newsletter's front cover, I needed to put "JACG" upsidedown and slanted.(ꟼVCG) Getting it is no problem, *but the screen and printout don't match!* Take a look at Figure 1.

Introducing a new version of an operating system always brings out the incompatibility bugs, and the new Megas are no exception. PP is one of the programs that stumbles (how fatally, I'm not sure.) Kazmaier is working to fix that -- the beta test 1.03 was apparently in use by mid-October. Anyone contemplating a Mega/PP combo should probably contact SoftLogik to find out when 1.03 will be available. And, those who like using lots of fonts, and don't want to wait for 2.0, might want to contact SoftLogik as well. Why? Because 1.03 is going to fix the 9 fonts and crash syndrome of 1.01. The release date for 2.0 itself will apparently be sometime at the end of this year, or the beginning of the next. The GEnie topic indicates that the upgrade will run $30.

ADVENTURES IN (IMG) SCANNING

As I reported last month, I ordered the IMG SCAN from Seymor-Radix. The scanner has arrived, and been attatched to the printhead and given a few hours of work.

In principle, the operation is very simple -- plug the small card into the cartridge port, tape the optic fibers to the printhead, put the picture to be scanned in the printer, run the program, and go. The basics are straight-forward. But getting good results is another matter.

The key problem is getting the end of the fibers the correct distance from the paper -- about a sixteenth of an inch. Too close, and the image will be too dark, too far away, too light. The second problem is adjusting the speed and image-size parameters. Figure 2 shows the main screen for the low-rez program. The reduce/expand bars are used to control the size of the scanned area, while the box in the left corner can be moved around to set which part of the image will be scanned. The indicators along the bottom control the conversion of the 256 gray-scale raw data, to the 16 gray-scale low-res format. The colors can be changed, and the four commands in the middle are straightforward.

The problems come in editing the scan parameters, one of the menu selections. The scanner supports Epson compatibles, but will allow the entering of the printer-specific codes for the vertical spacing. That's easy enough. But the documentation does not adequately explain the relationship between the parameters, and

how the scanner will react. Some samples, based on different printers, would be very useful, assuming that the print speeds and paper advance speeds were included.

So, in using the scanner, be prepared to spend some hours twiddling the parameters. Eventually, you may get something worth saving. Figure 3 is an example, taken from a photocopied page in the Disney Channel guide.

So far, I have talked about the low-rez mode. But what about the hi-rez, which, after all, is what most Publishing Partner users work in. Alas, the news is not good – so far, I have been unable to successfully scan anything (only a few hours spent, so far, admittedly). The problem is that the program does no dithering of the data, but simply chooses black or white, depending on the cutoff level. If you do not choose this level correctly, the picture will end up nearly all white, or all black. Worse, unlike the low-rez program, you cannot shift the level and have the program recalculate the image.

Conclusions? If you have a color monitor, and some time and patience, IMG SCAN is certainly worth looking into. But if you have only a monochrome monitor, I cannot recommend getting the scanner. Not until the hi-rez program is improved drastically.

ADVENTURES IN DRAWING (STYLUS STYLE)

Ever get tired of drawing with the mouse? It is a magnitude of improvement over the joystick, and perhaps about on par with using the Koala Pad with the 8-bit system. But how about something a magnitude better than a mouse (if you're willing to pay the bucks)? How about a graphics tablet from QMI and Mitsubishi?

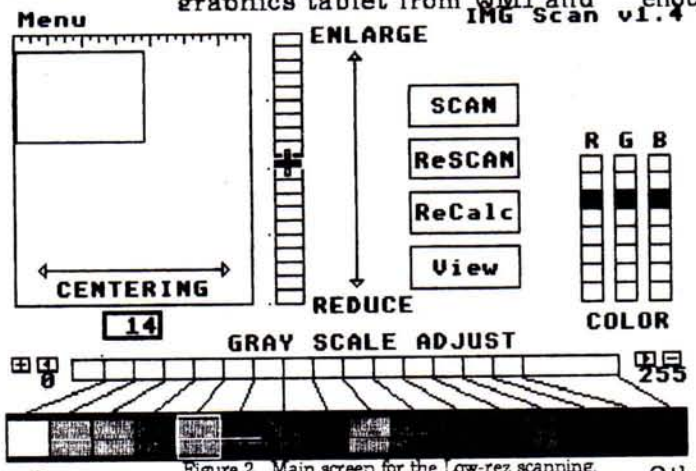


Figure 2. Main screen for the Low-rez scanning.

Well, several weeks ago, I saw the tablet at GEMINI, and, after arguing with myself, gave in to temptation. The tablet is 11x17, with an 8x12 drawing surface. It comes with stylus, power supply (another adaptor on the plug setup), RS-232 cable, and QMI's software. The RS-232 cable, interestingly enough, ends on the tablet side in a 9-pin, joystick-style connector. This suggests, that if the 3-foot cable isn't long enough, then it might be possible to use a joystick extension cord.

Figure 3. A picture scanned by IMG SCAN.

Figure 4. A befuddle computer user.

The software comes as an accessory, and automatically inserts the driver at power-up. The mouse and tablet are selected alternately, by pressing the ALT-SHIFT-CONT keys, or by selecting from the accessory. The section of the drawing surface to be used can also be set. (The tablet has a resolution of 250 lines/inch, so the drawing area can be quite small.) Other than these selections, the driver is completely transparent, and is supposed to work with any program. It certainly works with Degas Elite, as Figure 4 shows. That picture was first done on paper, in about 5 minutes, then the paper was taped to the tablet and copied into Degas, in about a half-hour total.

At the AtariFest show in DC, QMI was showing off the tablet (along with the other software). The tablet is an intelligent peripheral, having a dedicated processor inside, and can be set to several modes via dip-switches. Mitsubishi is apparently trying to break into the graphics tablet market with relatively low-cost tablets. (For comparison, the tablet available from a Maryland company for about the same price, has only a 6x9 drawing area.) So far, according to QMI, they are buying more for the ST than anyone else is, for any other computer.

As I said, it's wonderful to use, being much closer to drawing with a pencil. At $395 suggested retail, it's obviously not for everyone. But, if extensive graphics/CAD work is what you use the ST for, this device is definitely worth the investment.

Til next time!

PDG

Doug Van Hook - JACG

This month's Disk could be called, "THE PRINT SHOP'S GREATEST HITS!". The disk was assembled from the best utilities available for Broderbund's Print Shop. The remainder of this article appears in the documention, and was written by the SPACE LIBRARIAN:

CONTENTS (Front Side):

1. THE ICON SHOP. An outstanding Print Shop Graphic editor/creator with many advanced features. Refer to the accompanying instructions for details. (by Gregg Tavares, via the Genesee Atari Group in Flint, MI; G.A.G. Disk #149)

2. PACKTOOL.EXE. Compacts ("scrunches") or uncompacts ("unscrunches") Print Shop Graphics images (which are in DOS file format) for fast modem transfer or convenient "theme" storage. In ACTION. (by Dave Oblad, via G.A.G. Disk #97)

3. PSPIC.EXE. Converts Print Shop Screen Magic Images (the full screen type, not the small Graphic images) to DOS disk format. Also, PSPIC allows you to overlay this picture with mutiple Print Shop Graphics images. Read the following "DOC" file for details. (Ver 3.0 by M.G. Collins 10/85, via G.A.G. Disk #149)

4. SHOPTOOL.EXE. Use to view or to convert Print Shop Graphic images to DOS file format (or vice-versa). In ACTION. (by Dave Oblad, via CompuServe)

5. DOCREADR.EXE. Displays or prints out text documentation files. You're using it now! (by Jim Thompson via ANTIC PD)

6. DOS.SYS/DUP.SYS. Atari DOS 2.0 with a modified DOS.SYS file to support the Icon Shop.

CONTENTS (Back Side):

1. ARTSHOP.BAS. Converts Micro-Painter pictures to Print Shop Screen Magic images and vice-versa. Also formats disks (both types), renames, deletes, prints file listings of images on Print Shop type data disks. (by Forrest A. Blood, JACS, from the JACG Library)

2. BASICON.OBJ/BASICOFF.OBJ. Use to enable/disable the BASIC Language on XL/XE computers. Binary Load (DOS Option L) from DOS.

3. DOCUMENT.ATN. A BASIC program which prints out "DOC" files in a similiar fashion (ie, to screen or printer) as the DOCREADR.EXE program above. Allows user selection of the file to be printed/viewed. (by Tom Pazel, JACG)

4. FLIPPER2.OBJ. Use with DOS file formatted Print Shop Graphics images to flip, inverse, overlay letters and make other modifications to the picture. Built-in documentation instructions. NOTE: The Graphics' filename must have a ".PSF" extender. Also, when prompted for a filename (during loads or saves), be sure to type in the complete filespec, e.g., "D:MYPIC.PSF". In compiled BASIC. (by Jim Steele of G.A.G. via Disk #116)

5. PSFCONV.BAS. A BASIC program that will convert saved Print Shop Screens to DOS file format. (NOTE: Do not use with Print Shop Graphic images/pictures!) Created files are 63 sectors long. A preformatted DOS disk is required. (by Mike McCuen, from ANALOG #38, Jan 86, pg 95)

6. PSFLOAD.BAS. A BASIC screen load subroutine for use with pictures created with PSFCONV.BAS, above. (by Mike McCuen, from ANALOG #38, Jan 86, pg 95)

7. PSPRINT.OBJ. Prints out to an Epson/Star (or compatible) printer a listing (the picture + name) of all Print Shop Graphic images contained on that disk. Up to 70 pictures are conviently printed on each page - each in the proper highth/width ratio! Thus, the contents of a "packed" disk will require only 2 pages. Compiled BASIC and VERY slow, but worth the wait! (by Rick Harroun, via G.A.G. Disk #124)

8. PSUTIL.BAS. A BASIC utility for Print Shop Graphic images on Print Shop data disks only. Use to view, rename, alphabetize (and make a 5 column filename print out), or just print out 20 graphic images per page using Epson, Gemini or Prowriter type printers. Refer to the following DOCs for details. (by Tom Pazel, JACG; JC mods)

9.PS2VIS.BAS. Converts a Print Shop Graphic image into "VISUALIZER" format as a normal DOS type file. Use joystick to position the image anywhere on the 1760 byte VISUALIZER screen. (by Tom Pazel, JACG)

10. SHOPRINT.OBJ. Use to print out listings of the contents of Print Shop Graphic disks using Epson/compatible printers. Similiar to PSPRINT.OBJ, above, except the printed pictures are larger sized and slightly elongated. Two printout sizes available, however. (by unknown, via CompuServe)

11. Atari DOS 2.5. The DUP.SYS and RAMDISK.COM files are also included.

NOTES:

1. For best results, please "BOOT" the front side of this disk with BASIC disabled (e.g., turn on your computer while pressing OPTION; 400/800 users remove cartridge).

2. If you have a printer, you are encouraged to make printed copies of all documentation files (this file is named DOC.000; the others are: DOC.001, DOC.002, etc.) for convenient and immediate use as required.

3. Some of these documentation files were written by the SPACE Librarian based upon limited use of the subject program. Therefore, certain program features may be omitted, incorrect or not fully explained. Your corrections, additions or enhanced programs would be, as always, greatly appreciated!

This disk is J.A.C.G. disk number 129D. As always, the disk of the month costs $3.00 at the meeting, and $4.00 by mail. To order by mail, requests can be made in writing to:

MAIL ORDER LIBRARIN
Bret Callegari
306 Division St. Floor 2
Boonton, NJ 07005

◆

Where Is Our Support?

Nobody Likes A Winner

Steve Godun - JACG

Yes, another article dealing with the injustice Atari 8-bit computers have been enduring for the past few years. I'll just dive right in now and hope that this article sparks some enthusiasm (and maybe a few angry letters to software houses).

My primary argument for 8-bit support is that the software houses owe us. After all, many software houses, like Epyx, Electronic Arts, and Synapse got their foundations from 8-bit Atari support. And now, when Atari computer enthusiasm is higher than ever, they neglect us. Why? Nobody seems to know. And if the software houses know, they're not saying.

The most popular excuse that software companies use is "Atari piracy is the worst among all home computers." Ha! True, software piracy is rampant, but it's among ALL home computers, not just the Ataris. I've seen Commodore and Apple libraries filled with hundreds of pirated titles, yet support for these machines comes in droves. Atari computers (excluding the ST series; they have enough support) are simply scapegoats in the computer world. Even other, less powerful computers such as the Commodore Vic-20 have received more support even when they were discontinued! Injustice will never cease.

Another popular excuse is that "there isn't a large enough user base among Ataris." What?? Atari computers have as much of a user base, if not more than, C-64 computers. Once the price of the 8-bit computers and disk drives dropped to below that of a C-64 (which, in case you haven't already noticed, has had a higher cost than an equivelent Atari 8-bit for the past few years), people started buying them. Also, I pose this question to software houses: If there isn't a large enough user base of Atari 8-bit computers, then how can companies that primarily support the 8-bit, such as ICD and Supra, make their livings?

Another reason: "It's difficult to port a program from the C-64 and Apple to the Atari." Yeah, sure. I'm sure it's extremely difficult to port a program from one 6502-based machine to another 6502-based machine. Bill Budge was successful in porting Pinball Construction Set from the Apple to the Atari in a mere two weeks! I've done my share of Apple and Atari ML programming, and aside from graphics and sound, there is little difference between coding for a program.

Perhaps the bottom line here is that the Atari 8-bit computer is simply the scapegoat of the computer world. Why, I cannot tell you. The Atari 8-bit is the best computer you can get for the money. It costs less than 1/4th the amount of an equally equipped Apple //e, it is more powerful and easier to program than the C-64, and it has a very large user base. Without a doubt or a regret in mind, all I can say is that the software companies just don't care.

Before I close, I should point out the few good people in the Atari 8-bit support network. First, to Sarah and Bob Mutke, owners of Gemini Enterprises, who (contrary to rumors being spread around) are NOT going to drop support of the 8-bit Atari computers. And, of course, to Atari itself who is continuing 8-bit support with the XEP-80, SX-212, and XF-551 peripherals. I hope that this article will provoke some action among you who read this. Scan through an issue of Family Computing or Compute and find the addresses of the non-Atari supporting businesses, then sit down with AtariWriter and send them your complaints! Group efforts have been successful in the past; maybe together we can bring about more support.

GETTING LONGER LIFE FROM YOUR RIBBONS

Neil Van Oost, Jr. -JACG

About a year ago I purchased a reinking kit from Raven Systems Of Oakland,for my Centronics printer. The Inkfinity Kit -tm- sells for $5.00 and consists of two felt pads and a 30 cc (1 oz.) jar of ink. Refills of ink will cost you $3.00 plus shipping and handling. The ink comes in blue, black, brown, and red

Installation was rather easy and not too messy, although I would reccomend using gloves for the initial inking. Following the instruction sheet, you locate the take-up side of your ribbon cartridge and look for a flat place to mount the felt pad. The pad must be mounted were it will not interfere with the print head. Also it must protrude approximately 1/16 inch so as to allow the ribbon to make contact as it is being rewound. for mounting use two or three drops of Crazy Glue -tm-.

It took me about half hour to get the ribbon reinked initially. First I wrote a little program to make the print head travel back and forth with a minimum of printing. The object was to keep the ribbon in motion over the felt. The instructions say that the initial reinking will use approximately 30 to 40 drops of ink, and that is just about what I used. After that 3 to 4 drops of ink on the felt every 30 pages or so will keep your print nice and dark.

Rember to skake the ink up before each application. This is important as the ink contains oil,(as do all printer inks) which lubricates the print head.

A couple of weeks ago I tried to purchase a ribbon for my Star NX-10. The popularity of this printer is not the only thing that has risen over the past year. The cost of ribbons for it have gone from $6.00 and change to $8.00 and change. So last week I installed The Inkfinity Kit -tm- on it also.

With the number of posters and banners I print over the course of a year I figure I have saved the cost of 10 ribbons for my Centronics. Depending on the amount of printing you do the ink contained in The Inkfinity Kit -tm- should last several years. For more information, write:

Raven Systems Of Oakland, NJ
PO Box 253
Franklin Lakes, NJ 07417

NOISE from NOYES

D. B. Noyes - JACG

In the bargain area...I stopped by GEMINI, several ATARI 8-bit cartridges (new and in original pack) were available for $1.95 each. I, of course, picked up those that I didn't alreay have! I also noticed that GEMINI was carrying two new releases from MICRODAFT (JACG's own Eli Tomlinson's company) entitled "DROPZONE" and "SOLAR STAR". They are also advertised in a two-page spread on the inside cover of the October issue of ANALOG. Perhaps we can prevail on Eli to demo them at an upcoming meeting...8-bit Vice President, please take note!

I also saw (at GEMINI, where else?) the MEGA ST's. Not being a non-8 bitter, I dont't understand much about them...but they are there for those of you 16-bitters who are interested. My goal is to get a mega 8-bit machine, probably a pipe dream, but...one can hope!

A plug for the disk library. I hope that you have noted (as I have), the monthly meeting demonstrations of the Disks of the Month, by Doug Van Hook for the 8-bit, and Charlie Miller for the 16-bit machines. The quality of the PD (Public Domain, for the un-initiated) software in our libraries is excellent; as their demonstrations have indicated. Please avail yourself of the opportunity to get a lot for a little. Buy from the disk library every meeting!!!

A plug for the BBS. We have a well-run and maintained BBS (Bulletin Board System). Major components are a message base, a ZMAG base, and a file area (8 & 16 bit). If you don't have a modem, you can get a 1200 baud modem for well under $100.00. Not only can you keep up to date on th latest in ATARIDOM, but you can make extensive use of the file area for both up & downloads. Yes, the BBS has its "ups & downs"...couldn't resist it! And, best of all, there is a message base specifically reserved for Newsletter submissions...please submit to your hearts content!

==========
TRADING POST
==========

Trading Post is a service for JACG members who wish to sell or swap items of any type. There is no charge for this service. Material must reach the Editor by the 20th of the month to be considered for inclusion in the following month's Trading Post. No commercial services or items will be accepted.

>>>>>>>>><<<<<<<<<

Atari Fest, D.C.

BY Linda Peckham

Northeast New Jersey may have it's highway 80, but Washington, D.C. has its Beltway. On the 24th of October, I was reminded of this fact of DC life, as I sat in my car, traversing five miles of construction in the breath-taking time of one hour. (Not much was going on, of course; the problem was a one-lane bottleneck.) Nevertheless, the aggravation was not unworth it, for the chance of attending an all-Atari show.

The AtariFest was organized by Novatari, one of the D.C. area groups affiliated with the much-acclaimed *CURRENT NOTES* magazine. This was the third or fourth show held by this group, and was held at the Fairfax High School in Fairfax, Virginia on the 24th and 25th of October.

Like most shows, the DC show included a large display room and a seminar room. Besides these, however, there were also about a dozen classrooms taken over for specific interests, ranging from education to telecommunications, productivity, the Macintosh emulator, Midi/Music, and even a room for swapping /selling second-hand equipment.

Most of the activity, naturally, was taking place in the display room. Probably, every Atari dealer in the DC area was there, as was Atari itself, though without the famous striped tent. The major 3rd party companies represented were QMI, Migraph, Astra and Analog. Hybrid Arts, of Midi Maze fame (that is, 16 people playing against each other on 16 STs all hooked together through the Midi ports), was conspicious by their absence. (The DC show came two weeks after the Massachusetts show. This, coupled with the DC show's smaller (?) size, may have kept some companies from showing up.)

The big news on the 8-bit side seemed to be primarily the availability of the XE game system, and several new games. (There was no evidence of the 80-column card or of the new drive.) The game system drew the largest crowd around the Atari Corp. tables, with all of the kids trying out the "gun" that comes with the system. Other than the abovementioned no-shows, Atari had all of the new products there -- Laser Printer, Mega ST, XE Game System, and the PC Clone. While I didn't really see or hear that much, I did overhear Neil Harris (of Atari), tell someone that the Mega keyboards, did in fact fit in the 520s (and 1040s?), and that Atari was giving consideration to putting those keyboards into those two machines. (As production or upgrade, or both, I don't know. The Mega keyboard, and its potential for being put in a 520 or 1040, has been the focus of a fair amount of discussion on GEnie.)

On the ST 3rd party side, Migraph was showing off Easy Draw and Supercharger (their answer to Publishing Partner). Meanwhile, on opposite sides of the room, a kind of unspoken rivalray was going on, as to who had the better graphics tablet. QMI was showing off the Mitsubishi tablet, along with their other software, and giving people a chance to sit down and draw with it. (They also had a presentation in the seminar room.) Meanwhile, a Maryland company was showing off a more expensive tablet, with an artist offering to draw characatures for several dollars. At another table, the president of Astra was showing off the HD+, a 20.8 Mbyte Hard Disk plus double-sided Floppy unit. The main point stressed in his presentation is the unit's reliability (1 year warranty). Actions were more impressive than words, as he moved his 14-month old unit around, setting it on its side, putting it on top of the computer, and all the while, the hard disk was being accessed by a slide show. An impressive demonstration. (The Astra 5 1/4" drives were also on display, but I am not sure that they were hooked up.)

New software was also on display, and for sale -- *SPECTRUM* (the 48 colors per line paint program from Antic), a new terminal program, the various CYBER programs, and *WORD PERFECT*. I also glimpsed, but did not have a chance to check out, a demonstration of *DRAFIX 1*, one of the new CAD programs coming out. One interesting feature that I did note was that, instead of using the GEM drop-down menus, this program used 1 or 2 extra horizontal menu bars, which kept the screen work area clear.

Away from the press of the crowd, other rooms held the seminars and the hands-on demonstrations. David Small, of Magic Sac fame, gave a talk to a well-filled room about the development of the Magic Sac and the Translator, which will allow the ST to read Macintosh disks directly. (The Translator shipping date has slipped to November, due to a shipment of incorrectly labelled chips, which, after being soldered into the Translator boards, promptly blew up a number of the processors. To say that Data Pacific is unhappy with that particular supplier, is putting it mildly.)

Word Perfect was demonstrated in the Productivity room during the afternoon – or, rather, a demonstration was attempted. Having come out so recently, the person had not had time to get familiar with this top of the line word-processor. And it is, unquestionably, intended for the power user. Contained on six disks, this program is supposed to include virtually every feature a writer might want or need: Thesauras, Spell-Checker with 115,000 word dictionary, Outliner, automatic footnotes, indexing, etc, etc. One to five columns in a WYSIWYG format, and using GEM. And support for 200+ printers. The discounted price apparently will be about $230.

There was certainly more to see, and hear; but I ran out of time on Saturday, couldn't go Sunday, and I'm running out of room here. Suffice it to say, going to an Atari only show can be a fun (and money-spending) experience for any Atari Enthusiast.

Maybe we could have one in New Jersey someday.

MY FIRST ROBOT

by Donald Forbes - JACG

I saw my first live robot last month. I don't mean the toy that Art Leyenberger showed us on the grand piano a couple of years ago, or the rolling platform that Doug Van Hook and his buddies demonstrated some months ago. This was a real industrial robot performing in an industrial setting.

For years we have been using computers to replace human brains in various ways, and the results have been pretty good. When it comes to using computers and machines to replace human bones and muscles, then it becomes an entirely different game, with a completely different set of rules.

The robot reminded me more than anything else of a three-foot monkey. Sit him in a chair, tie one arm behind his back, and then blindfold him. The monkey could then do pretty much what the robot could do. Of course, you could probably buy a dozen monkeys for the price of this robot, but that is a different matter.

The reason the robot reminded me of the monkey is that the limitations of the robot were similar to those of the robot. Like the monkey, the robot could turn in his seat to the left or right in an arc of about 320 degrees (a little less than a full circle), which is about as much as the monkey could turn.

The monkey would be able to reach upwards about two feet into the air, which was about the reach of the robot. And he could reach anything within about a two foot radius of his seat, or about the reach of the robot.

Our monkey is blindfolded, so he cannot see anything. He must be told where everything is. Furthermore, he cannot move his hand from point A to point B in a straight line if there is an intervening obstruction. If there is an obstruction, he has to be told to move his hand in a series of steps so as to avoid it.

Perhaps the most interesting similarity is the anthropomorphic construction of the machine (this is a ten-dollar word from the Greeks meaning "human" and "shape"). This robot was constructed with a torso from which there rotated an upper arm, extending to a forearm which ended in a wrist, which held a hand capable of two motions, rotation and yaw. The hand, like a human hand, could rotate (like turning a dial on the TV set) and yaw (as when you wave goodbye).

The resemblance didn't end there. The robot control manual referred to the parts as the "arm" and the "elbow" and the "wrist" and the "hand."

Where did all this happen? Well, if you were at the August meeting, you may remember that I got up and announced that there would be a meeting of the Forth chapter at Rutgers, New Brunswick, and that you were all welcome. The topic of the meeting was to be "Robotics in Forth," to be held at the College of Pharmacy on August 27.

Professor Nick Lordi demonstrated his robot in the pharmacology lab for us, and showed us how he has been able to use Forth to create a robotics control language.

The robot was what is probably a standard industrial robot (there was a small square aluminum sticker underneath it with the word MITSUBISHI)and it was painted in a bright orange color. The robot was bolted to a heavy metal table and wired to a black box with a one-foot cubic capacity. The black box in turn was cabled to an IBM PC with a hard disk.

The aim of the demonstration was to show us how commands entered at the keyboard could be used to control the movements of the robot. The demonstration itself was designed for the pharmacology students to show them how routine processes can be automated in the laboratory. The students end up as pharmacists and must know how to combine the ingredients that are specified in a doctor's prescription.

In this instance, the objects on the table (besides the robot) were: (1) a rack holding a set of test tubes, (2) a spectrophotometer (which "held the tube up to the light") in which the amount of light passing through the tube registered on a meter, and (3) an empty rack to hold the processed test tubes.

In this exercise, the robot was expected to grasp a test tube, insert in into the light reader, and then remove the tube and store it in the empty rack. This is a task that any child could do, but it was a new experience to see it done by a machine. Dave Green, our tape librarian, was there and he, too, seemed to be impressed.

Nick first gave the robot the command NEST. And, believe it or not, this is what the robot did; like a bird about to go to sleep, the robot curled inself up into a compact unit.

Nick then gave the command EXTEND. The robot slowly and dutifully moved his arm out in front above the light reader. When Nick gave the command RETRIEVE the robot rotated his hand to the left, positioned it above the test tube, reached down, and clamped his two claws around the tube. (The tube, of course, had been positioned ahead of time in the exact spot where the robot would be told to find it.)

At the next command the robot moved from the waist to position the tube above the light reader, lowered it into the hole, pushed the tube down, and then reached for and closed the door to the light reader (to block out the overhead light). The dial on the reader showed a number (in the 90s) which was picked up by the computer.

reader, picked up the test tube, and then swung around and carefully placed it in the empty rack.

Everything was done precisely and methodically. Nick had inserted several delay loops because he was still testing the system. But all the steps worked because everything had been positioned beforehand in exactly the right location.

One of the principal problems of robot control is to specify the coordinates of all the motions. You can use regular Cartesian coordinates in three dimensions (x, y, and z), or cylindrical coordinates (distance and angle from the central pivot, and height above the table). The robot comes with a hand-held control box and an LED display so that the arm and hand can be moved in real time. And the control box gives you a choice of more than one set of control coordinates.

The reason that Nick could position the robot at any specified point is that he cheated. The PC was programmed to display at all times the exact coordinates of the arm. Once he had the coordinates, he could then position the test tube and light reader exactly under the hand. It was all done with mirrors.

Nick was planning to give us a copy of his Forth code that he developed, but his office copier was on the fritz that day. He will give you a copy if you call him (338-9363); he lives in Bloomfield at 19 Barnett street.

I asked Nick what were the economic reasons for the use of an expensive industrial robot (Rutgers is a state university) in a pharmacology lab, besides the obvious one that it could handle hazardous substances.

One of the principal reasons was that it could automate routine tasks. If a set of test tubes had to be analyzed, then the job could be turned over to the robot at the end of the day, and the results would be available in the morning.

Another important reason was that lab technicians can get sloppy or can not do tasks as precisely as a robot. For example, a given step in a procedure may require an exact number of seconds. Or a procedure may involve several steps with precise timing requirements. Or a procedure may involve multiple steps that must be done in exact sequence one after another. Human operators may make mistakes that a robot never would.

Why use Forth to control this robot? As everyone knows, Forth was developed originally for process control (moving astronomical telescopes). All commands to this robot are in the form of text strings, which Forth is well equipped to handle.

The other important reason is that, by developing his own robotics control language, Nick has complete control over his slave. If you go into the marketplace to buy a robot to do a certain task, you get the robot your mind and want to have the robot do another task, then that involves a new program and you have to hire a programmer and wait for him to write a new program and then you have to pay him. All that can be bypassed if you write your own code.

What are some of the constraints? The first one, of course, is that the robot is blind so you have to be careful where you move the arm. Nick almost damaged one of the claws when he told the robot to NEST without realizing that the hand would come down over the light reader.

Another constraint is that this robot is only designed to lift about one kilogram (two pounds). All the movements are created by belts moving on pulleys, so an overload would probably strain the belts and cause overheating. The robot, of course, is stupid so that if you give it an impossible command (such as forcing a nail into the table) it will continue to try to execute the command until something breaks.

What about the future of robots. Our generous disk librarian, Sam Cory, gave me a copy of Robotics World for July. The magazine says that, according to a U.S. Department of Commerce report, capital spending cutbacks in the auto and electronic industries -- the largest users -- could lead to a 20% dip in new robot purchases before the year end, but that the U.S. robot market will increase sharply in the next decade. More than 30 percent of domestic robot consumption is from imports, with Japan supplying 80 percent. Future success in the U. S robotic market appears to lie with the supplier that can provide a complete turnkey system to meet users' needs.

A French company has developed a hand-sized robot with a claw that can walk on vertical, flat or circular surfaces with magnetic feet. It can carry loads up to 6 pounds, and moves at about 10 inches a minute by alternately powering and releasing two pairs of electromagnets. Probable uses include inspection and repair of large piston and turbine engines, and checking for cracks in nuclear power equipment such as generators, pumps and valves.

The University of Western Australia has shown that 80 percent of the wool of merino sheep can be successfully removed by a robot, and they are now working on the other 20 percent.

Today there are about 26,000 robots at work in America. Chrysler Corp., when on the brink of bankruptcy, made a commitment to robotics. According to executive manufacturing VP Richard E. Dauch, this commitment played a major role in changing Chrysler from a $10 million loser per day to a $10 million profit-maker per day, or a $20 million per day swing in the launching of a new product.

Perhaps there is a moral here.

COMPUTER SYSTEMS CONSULTANTS, INC.

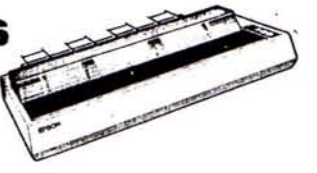
inc.

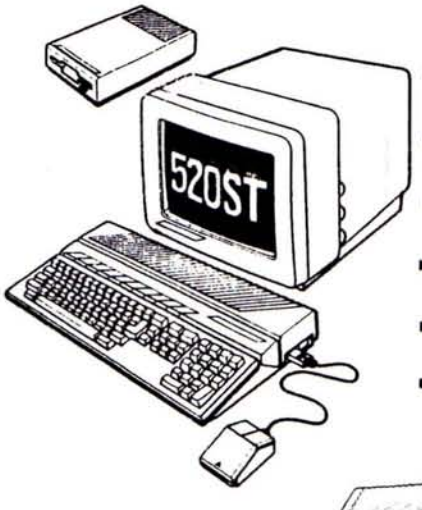
Box 873, 897 U.S. RT. 130
Hightstown, N.J. 08520
(609) 448-8888/9

BEFORE CALLING US, CALL AROUND
THEN CALL US FOR LOWEST PRICE!

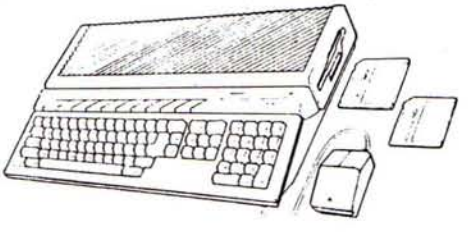
We Are A Certified Atari/Epson
Service Center/Dealer

Low Overhead = Low Prices





- 520 ST color or mono
- 1040 ST color or mono
- ALL EPSON Printers
- ST Software
- Peripherals
- Cables, etc.



visa-m/c
9 am - 5pm
mail orders

JACG MEMBERSHIP APPLICATION

DUES: US (inc. APO, FPO, etc.), Mexico & Canada $25.00
First Class mailing of Newsletter, add $6.00
Foreign Subscriptions (US Currency) $30.00

MAIL TO: ROBERT P. MULHEARN
8 CRESCENT ROAD
PINEBROOK, NJ 07058

RENEWAL____ NEW MEMBER ____ FORMER MEMBER _____

NAME_____MEMBER # _____

ADDRESS _____

CITY/STATE/COUNTRY _____ZIP CODE_____

HOME PHONE NUMBER _____

DATE _____ 1ST CLASS MAIL _____ 3RD CLASS MAIL _____

NEW TO THE ATARI?

NEED A LITTLE HELP?

MAYBE A JACG BIG BROTHER CAN HELP

GIVE US A CALL
201-469-6190

MEMBERSHIP RENEWAL

Take a moment and look at your mailing label on a recent issue of the JACG Newsletter. Check the upper right hand corner. This is the month/year when your membership expires. Try to renew at least one month early. This helps us keep our bookkeeping in order and avoids your missing any issues of the Newsletter.

There are two easy ways to renew:

1. Fill out a membership renewal form in the front lobby before our monthly meeting and present it with $25 (in cash or check) to the Treasurer. Add $6 for first class mailing of the Newsletter.

2. Copy the information on your mailing label and send, with your remittance to the address listed above.

CHECK YOUR LABEL TODAY!!!

NEWSLETTER ADVERTISEMENTS

RETAILERS - MANUFACTURERS - VENDORS

How would you like to reach a targeted audience of over 700 ATARI computer users? This newsletter has a press run of 800 per month, is read by members and non-members alike, and is sent to over 60 other ATARI User Groups across the U.S. and in several other countries.

Advertising is available on a first-come and space-available basis. Camera-ready copy, accompanied by payment, must reach the Editor by the 20th day of the month preceding publication. JACG reserves the right to make decisions concerning the placement of ads within the Newsletter and editing or rejecting advertisements deemed unsuitable.

ADVERTISING RATES

Full page (7-1/2 x 9) ...$48.00
Half page $25.00
Quarter page $18.00

Discount rates available on request. Contact:

Gary Goraki
313 Sheridan Avenue
Roselle, NJ 07203

JACG
JERSEY ATARI COMPUTER GROUP
8 CRESCENT ROAD
PINE BROOK, NEW JERSEY 07058

BULK RATE
U.S. POSTAGE
PAID
PINE BROOK, NJ
PERMIT # 56

SAN LEANDRO COMPUTER CLUB
c/o SLCC NEWSLETT.EXCHG.
P.O. BOX 1506
SAN LEANDRO, CA 94577

JACG NEWSLETTER - VOL. 7, NUM. 9

NOVEMBER 1987

JACG EXECUTIVE COMMITTEE

PRESIDENT

THOMAS PAZEL
70 Mitchell Road. T-7
Hackettstown, NJ 07840
201-850-9017

VICE PRESIDENT, 8 BIT

DOUG VAN HOOK
40 Meadow Lane
Clifton, NJ 07012
201-472-0637

VICE PRESIDENT, 16 BIT

THOMAS SHOOSMITH
93 Patricia Avenue
Colonia, NJ 07067
201-388-5952

TREASURER

SHREE VANDENBERG
220 Garfield Avenue
Plainfield, NJ 07060
201-753-2416
201-373-6789 Ext. 31

EDITOR

DAVID B. NOYES
3 Ann Road
Long Valley, NJ 08854
201-852-3165

SECRETARY & MEMBERSHIP

ROBERT P. MULHEARN
8 Crescent Road
Pinebrook, NJ 07058
201-575-0067

LIBRARIAN

SAM CORY
P.O. Box 7
Towaco, MJ 07082
201-334-4443

INTERNATIONAL CHAIRMAN

BARRY WEISSMAN
125 Runyon Avenue
Piscataway, NJ 07853
201-885-5161

MAIL ORDER LIBRARIAN

BRET CALLEGARI
306 Division St. Floor 2
Boonton, NJ 07005

DISK LIBRARIANS

Dave Green
Doug Van Hook
Charles Miller (ST)

PRESIDENT EMERITUS

JOSEPH S. KENNEDY
126 Jupitor Street
Clark, NJ 07066
201-381-5559

ADVERTISING & SALES

GARY J. GORSKI
313 Sheridan Avenue
Roselle, NJ 07203
201-241-4554

BBS CO-SYSOPS: Mark Knutsen, Gary J. Gorski, Thomas Shoosmith

The Jersey Atari Computer Group (JACG) is an independent, informal organization of ATARI computer users. It is not affiliarted with ATARI or any other commercial enterprise. Opinions expressed in this publication reflect only the views of the individual author, and do not necessarily represent the views of JACG. Material in this Newsletter may be reprinted by other Atari User Groups, provided the author (if applicable) and JACG are given credit. Only original work may be reprinted. Questions concerning reprinting should be addressed to the Editor.